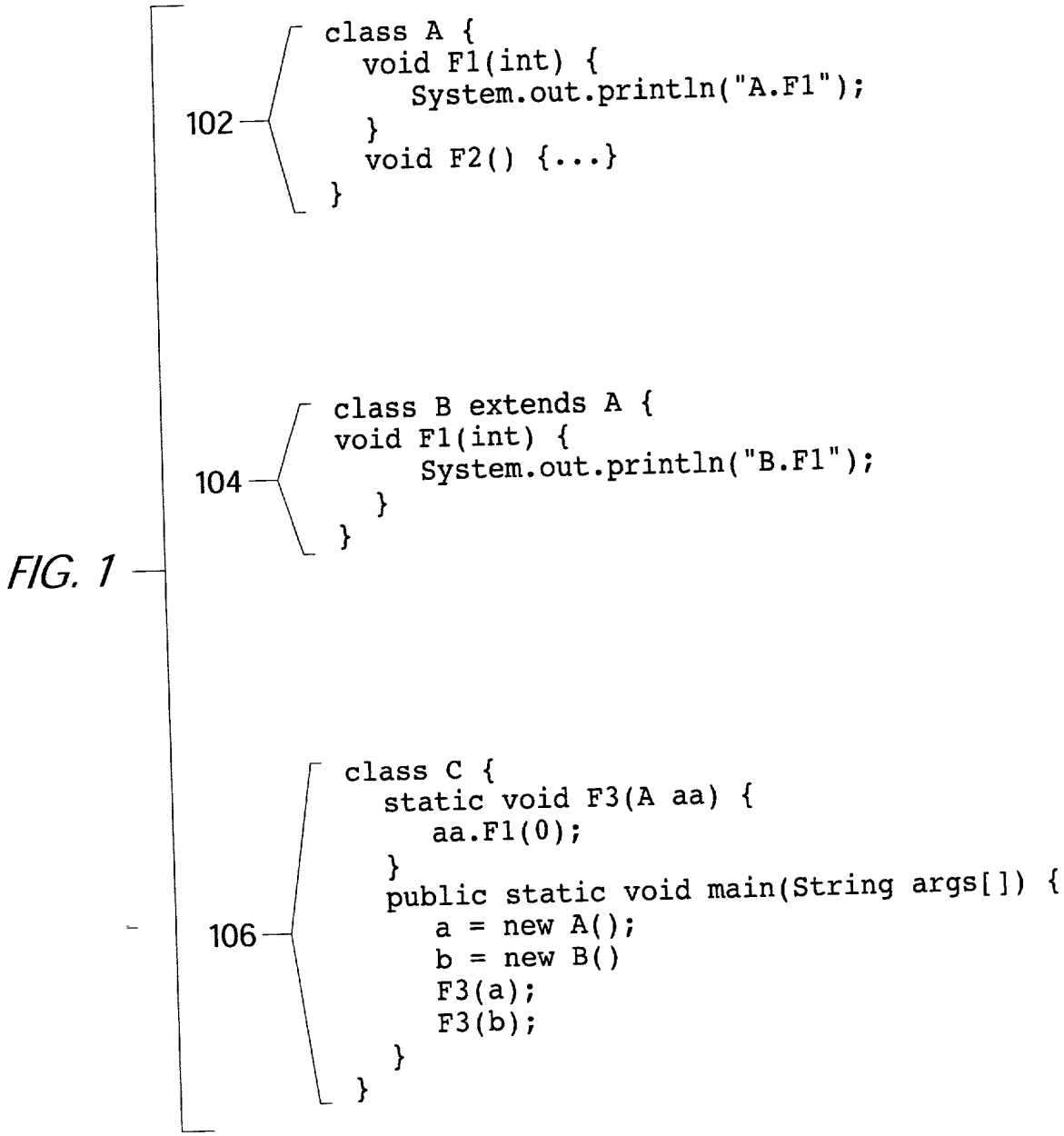
102
class A {
void F1(int) {
System.out.println("A.F1");
}
void F2() {...}
}
104
class B extends A {
void F1(int) {
System.out.println("B.F1");
}
}
106
class C {
static void F3(A aa) {
aa.F1(0);
}
public static void main(String args[]) {
a = new A();
b = new B()
F3(a);
F3(b);
}
}

FIG. 1

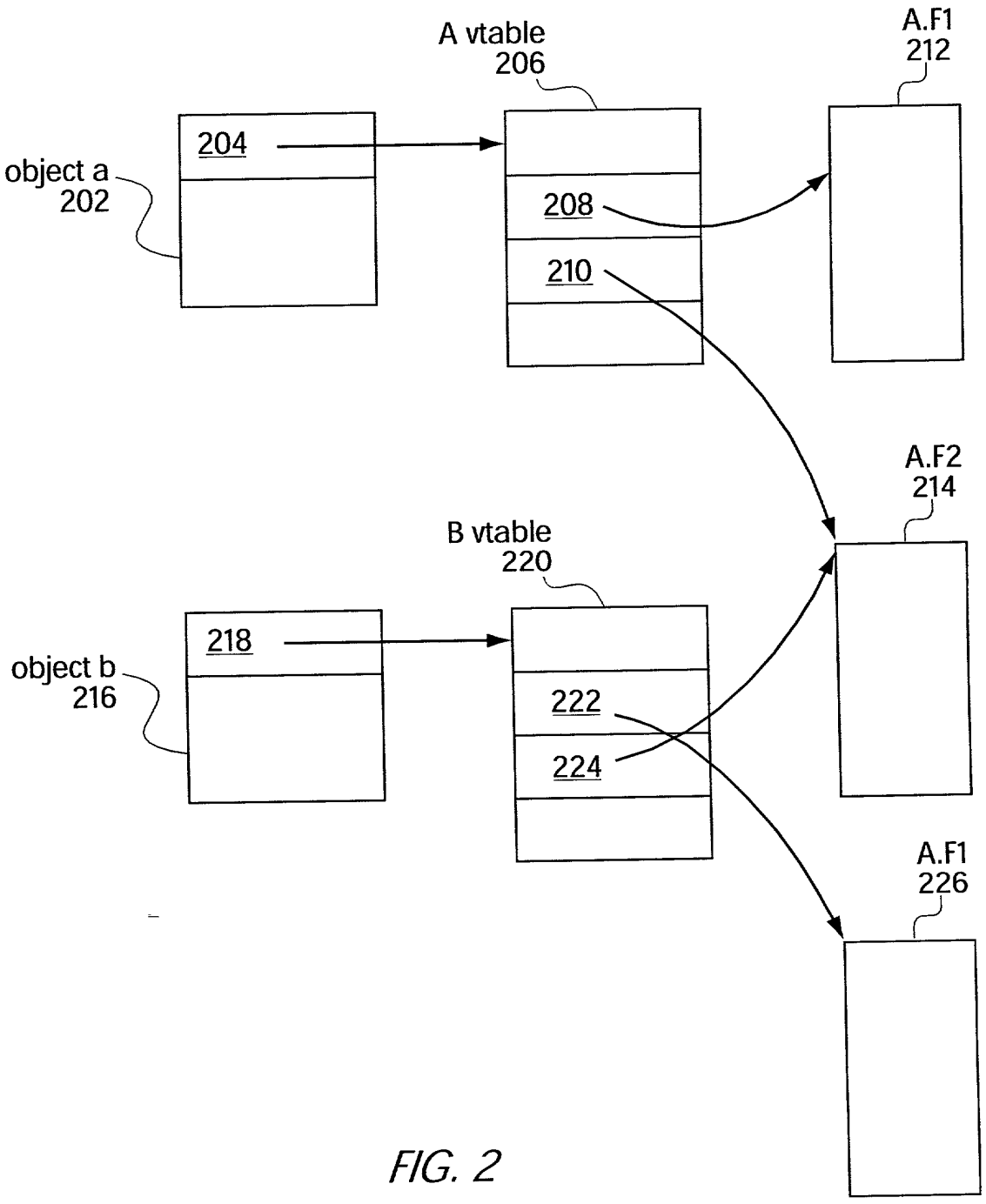
A vtable
206
A.F1
212
204
object a
202
208
210
A.F2
214
B vtable
220
218
object b
216
222
224
A.F1
226

FIG. 2

FIG. 3

302
```
interface D {
     void G1 (int);
     float G2(float, int);
}
```

304
```
interface E{
  void G1 (int);
  void G3 (int);
}
```

306
```
class F implements D {
  void G4();
  void G1() {
     System.out.println(F.G1);
  }
  float G2(float, int);
}
```

308
```
class G implements D, E {
  void G1() {
     System.out.println(G.G1);
  }
  void G3(int);
  float G2(float, int);
}
```

FIG. 3B

310
```
class H {
  static void G4 (D d) {
     d.G1(0);
  }
  public static void main (String args[]) {
     f = new F();
     g = new G();
     G4 ((D)f);
     G4 ((D)g);
  }
}
```

TOOT60 "9029860

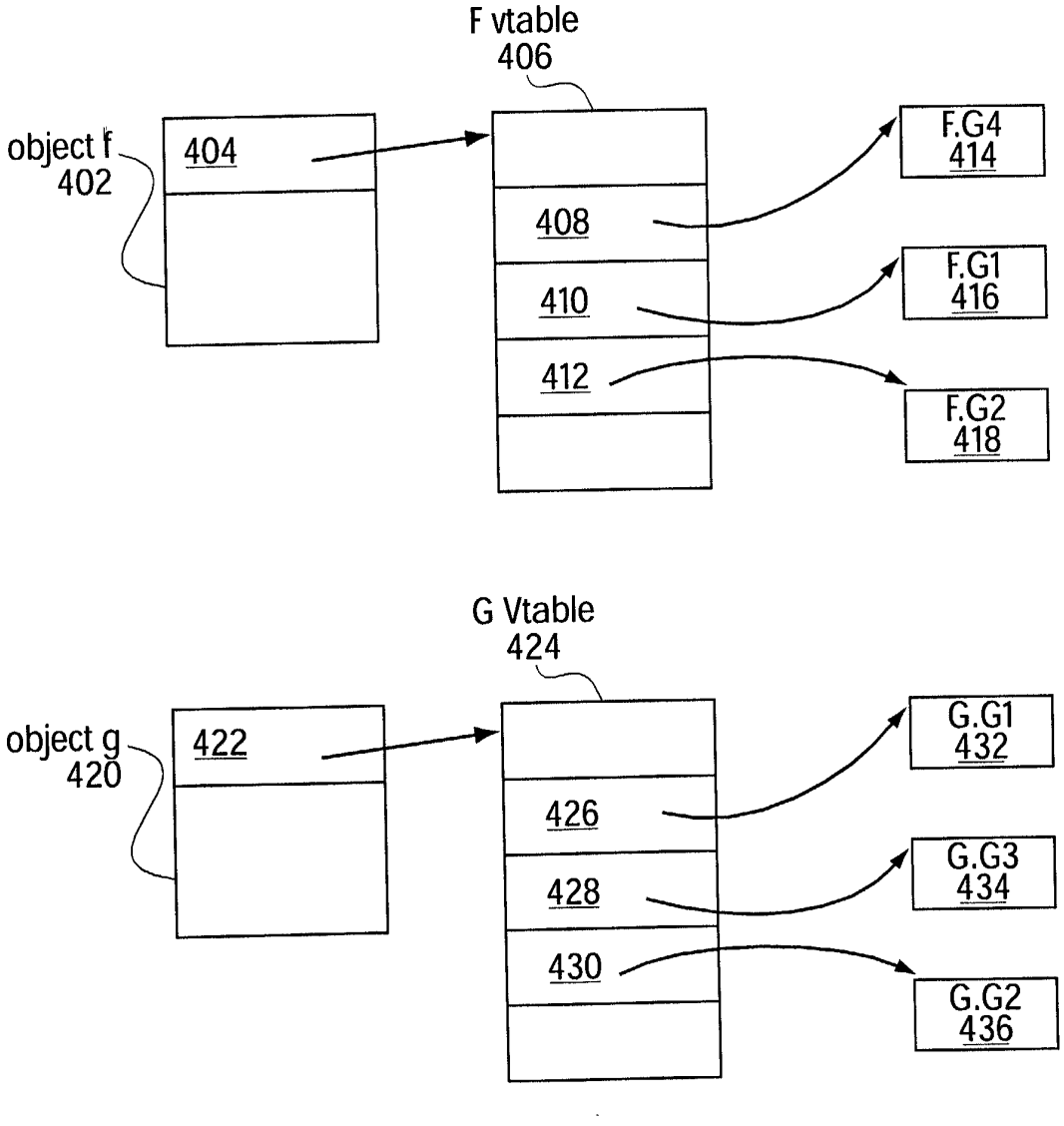

FIG. 4

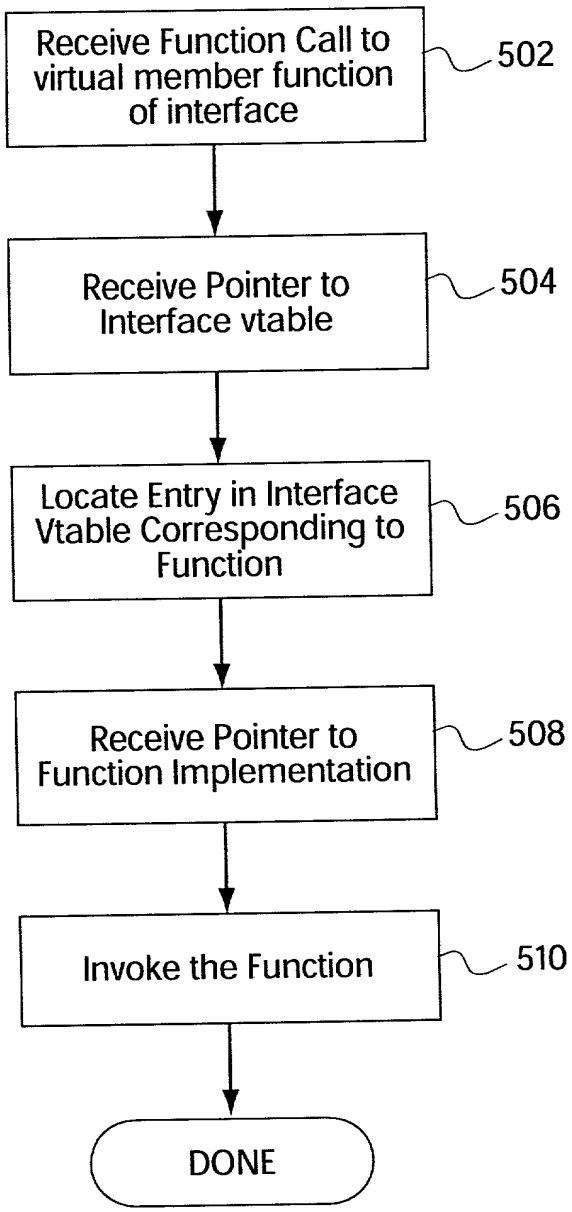
Receive Function Call to virtual member function of interface
502
Receive Pointer to Interface vtable
504
Locate Entry in Interface Vtable Corresponding to Function
506
Receive Pointer to Function Implementation
508
Invoke the Function
510
DONE

FIG. 5

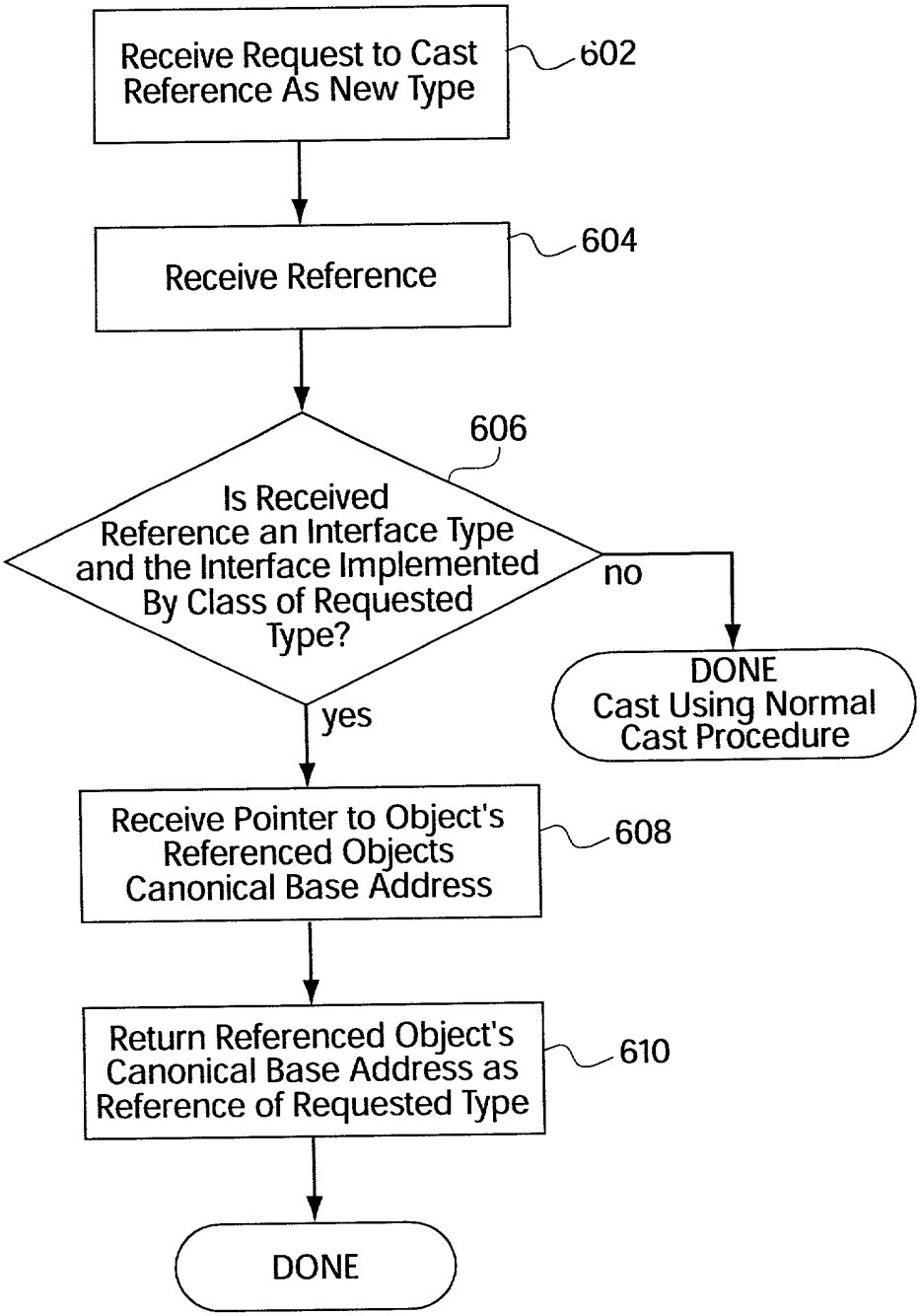
Receive Request to Cast Reference As New Type
602
Receive Reference
604
606
Is Received Reference an Interface Type and the Interface Implemented By Class of Requested Type?
no
DONE Cast Using Normal Cast Procedure
yes
Receive Pointer to Object's Referenced Objects Canonical Base Address
608
Return Referenced Object's Canonical Base Address as Reference of Requested Type
610
DONE

FIG. 6

T00760 "30296860

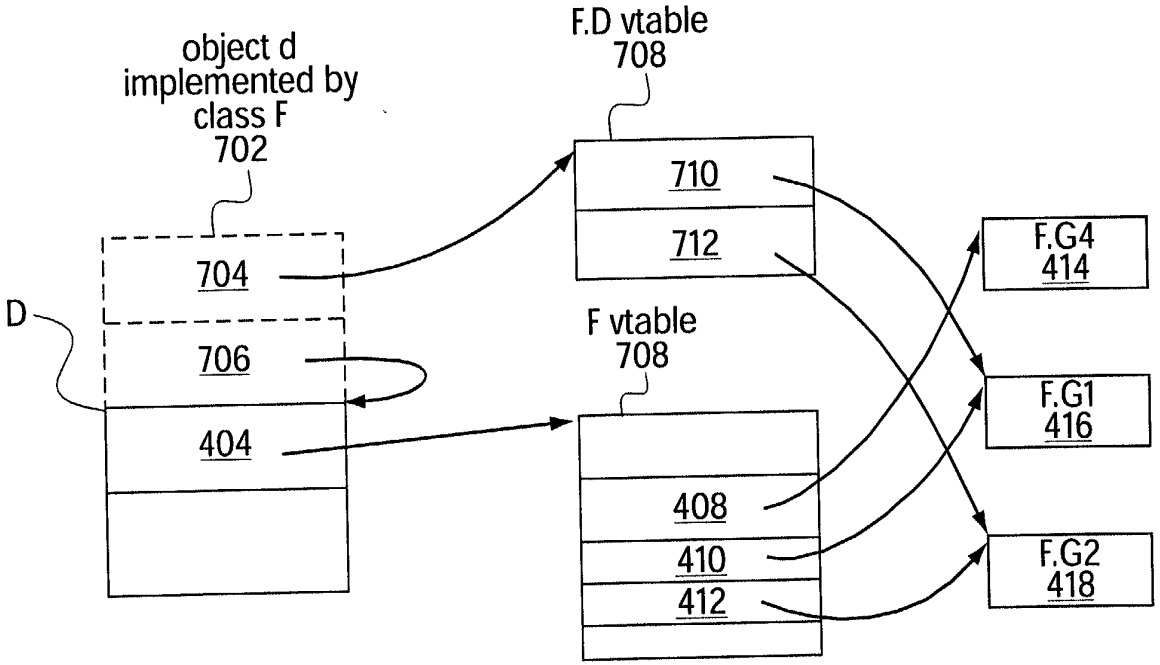

FIG. 7a

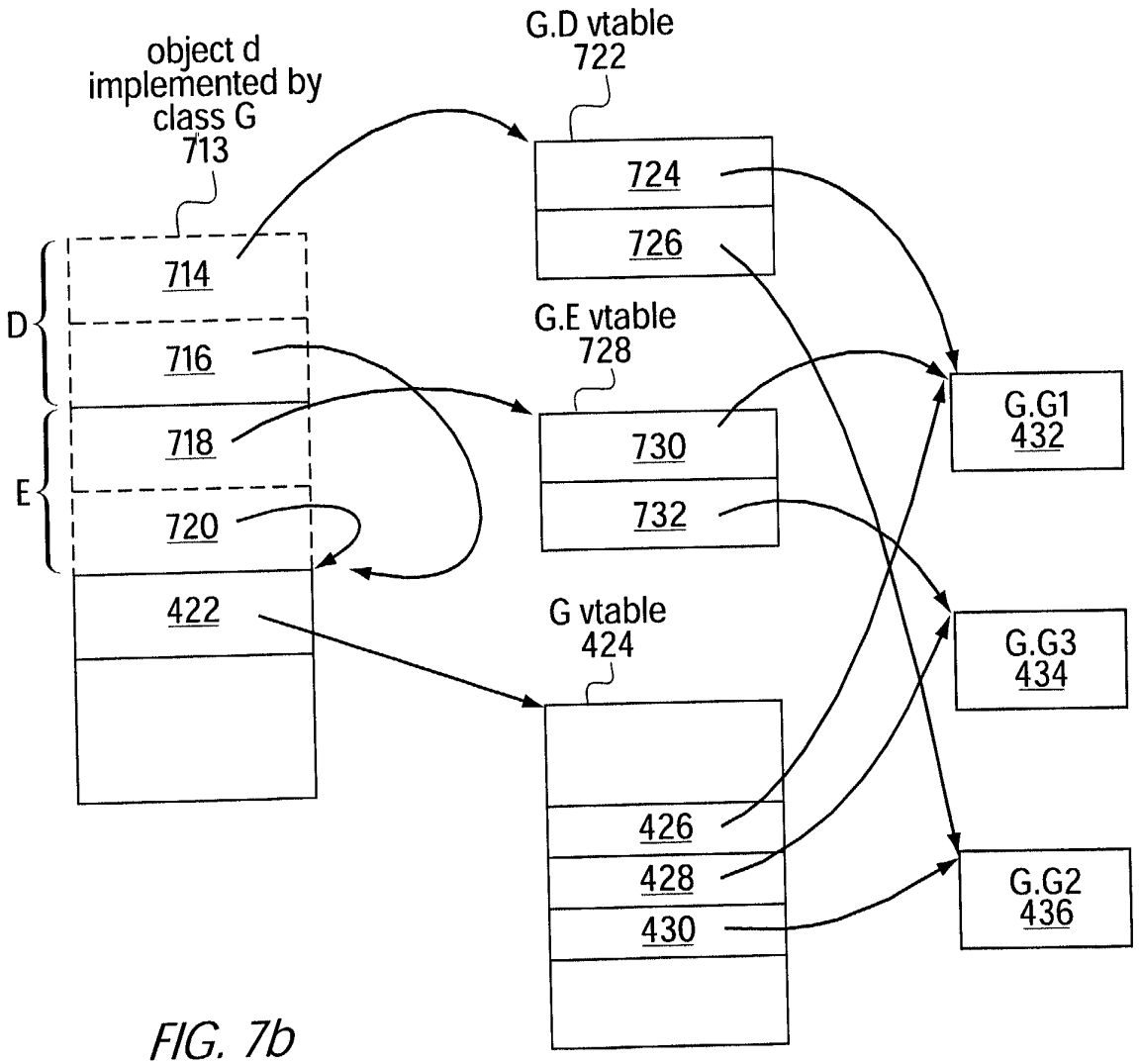
object d
implemented by
class G
713
G.D vtable
722
724
726
714
D
716
G.E vtable
728
718
730
G.G1
432
E
720
732
422
G vtable
424
G.G3
434
426
428
G.G2
436
430

FIG. 7b